Expanding the Geo URI (RFC 5870) to incorporate geocodes

Peter Krauss[1], Thierry Jean[1], José Damico[2], Everton Bortolini[1,3], Rui Pedro Julião[4], Alexander Zipf[5], Alan C. Alves[6]

ABSTRACT

The Geo URI (internet standard RFC 5870) is a compact sequence of characters that identifies a geographic location, with the syntax that starts with the 'geo' prefix followed by latitude and longitude coordinates. However, there are, nowadays, more compact and human-readable conventions to express the same geographic location, such as Geohash or OLC (Open Location Code) geocodes. This proposal describes a simple extension for the “geo:x,y” part of the schema, the sub-schemas “geo:x” (where *x* is a geocode) and “geo:type:x”, in order to include new syntactic and semantic conventions in the Geo URI.

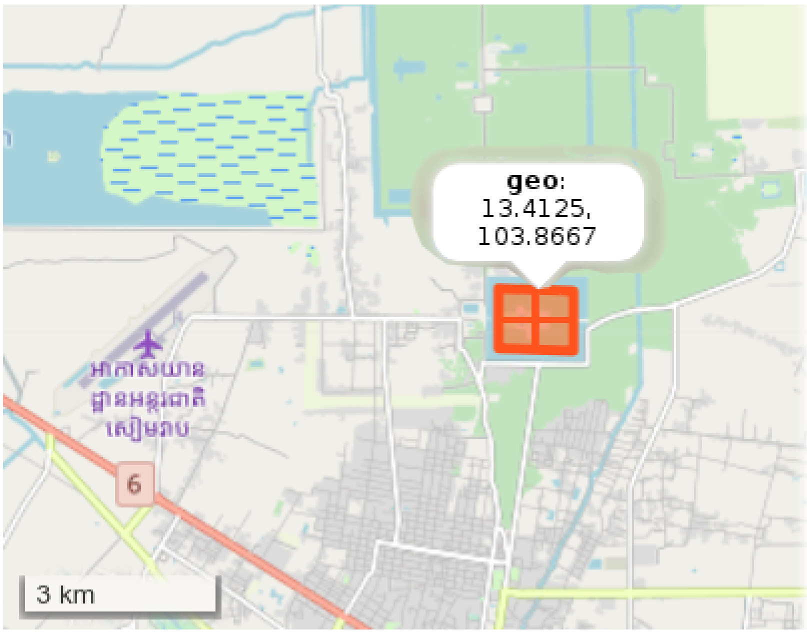
key-words: geocode, Geo URI, interoperability, geographical linked data, fine-grained postal code, internet protocol, point locator.

`art2-geoURI` v0.4.8 of May 7 '2019

[1] OpenStreetMap Brasil Community. ✉ peter@openstreetmap.com.br or thierry@openstreetmap.com.br.
[2] SciCrop.com
[3] Department of Geomatics, Federal University of Parana, Curitiba - PR - Brazil. ✉ evertonbertanbortolini@gmail.com
[4] Universidade Nova Lisboa, Dep. de Geografia e Planeamento Regional da NOVA FCSH, Portugal. ✉ rpj@fcsh.unl.pt
[5] GIScience Research Group, Heidelberg University, Germany.
[6] Municipality of Itanhaém, SP, Brazil.

SUMMARY

INTRODUCTION



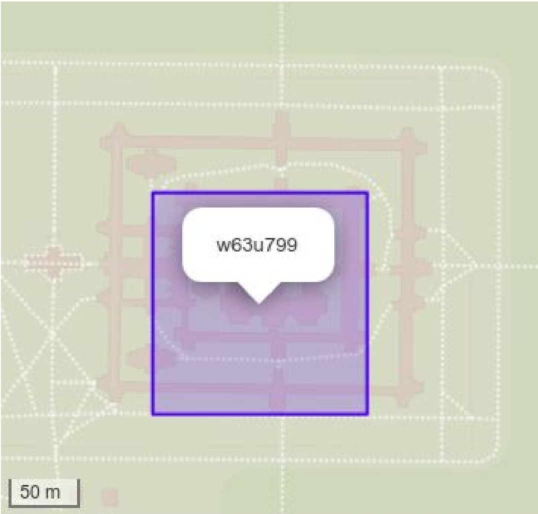
The internet standard RFC 5870 specifies a Uniform Resource Identifier (URI) for geographic locations, using the 'geo' scheme name. It is easy to express in a link or metadata, and easy to interpret, or to delegate, e. g. redirecting to OSM.ORG.

Eight years after its publication (2010), it can not be said that it was a public success when compared to other protocols, in terms of number of 'geo' client applications supporting it, or number of ‘geo’ URIs that we find on the Web. But it was certainly a great technological achievement as it provided the appearance of GeoJSON (RFC 7946) in 2016 and brought to Web standards the WGS-84 default.
Here is the first example given in the RFC 5870 document:

```
geo:13.4125,103.8667
```

This is the central point of the temple Angkor Wat. The same point can be identified by the geocodes Geohash “w63u799mm” (with ~5 m of precision) or OLC (Open Location Code/PlusCode) “7P55CV68+XM”, with ~10m. The aim of this proposal is to express these kinds of geocodes with the following 'geo' URIs:

```
geo:ghs:w63u799mm
geo:olc:7P55CV68+XM
```



The three `geo` expressions results in near same string-lengths, but the relevant information encapsulated into the geo expression, that informs the location and must be human-readable, is reduced from 16 characters of the “latitude,longitude” coordinates, to 9 or 11 characters of geocodes. The reduction is a gain for human memorization and readability.

More reduction is possible, when we do not need a point but a point-neightbout. Let’s go back to the Angkor Wat example. The main ruins are in a ~25000 m² area, while the whole temple covers an area of 1.63 km². Therefore, we can represent the ruins location by the “w63u799” Geohash cell (illustrated), with ~23000 m² of area, that can be approximated to a square with sides of ~150 m. It is a shorter code (7 characters instead of 9) because it discards some “non-used precision”, and it corresponds exactly to the user needs, a box with ~150 of side.

The advantages of geocode (over coordinates) are
their **shortness** and the fact they **express the area size (or precision)**.

Preprocessing

When, for a given application, the geocode is not directly consumed, the URI can be preprocessed, returning a URI with coordinates representing the same location. The size in the geocode can be converted to the location uncertainty parameter (*u*) of the 'geo' URI. For instance, the illustrated Geohash `geo:gsh:w63u799` can be expanded to something as `geo:13.4125,103.8667;u=50` or other better calculated value for *u*.

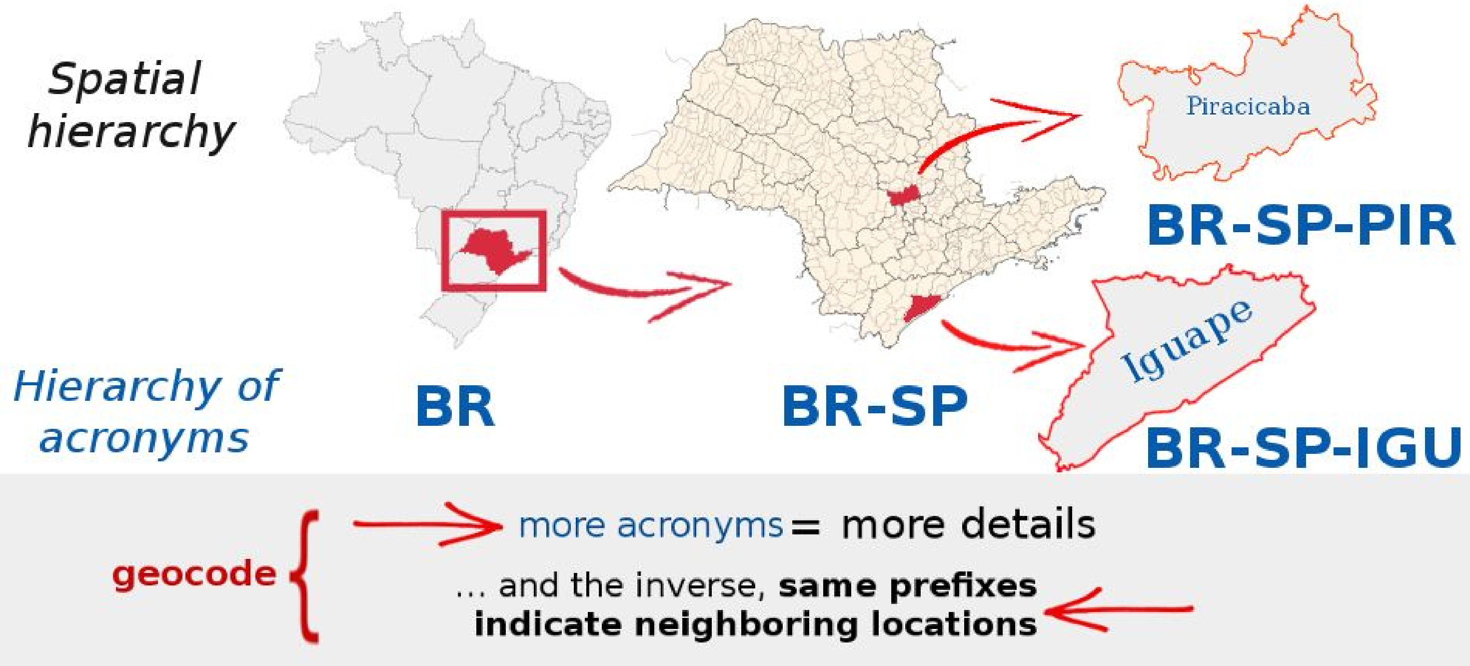
In software architecture terms, this allows the geocode to be used as a "second layer" metaphor, which backs to the ordinary 'geo' scheme by a simple translation procedure, also described in this article.

Mnemonic prefix for jurisdiction

There is no international standard, such as ISO standard, for a consensual geocode system. Geohash, OLC, and all others are niche specific, none of them have been widely used. But there is a well-known ISO code that identifies geographic areas, the ISO 3166-1 alpha-2, also called "country code". We suggest that ISO country codes be "mnemonic prefixes" in the geocode, which provide a geographical context for their suffix, and, mainly, determines the jurisdiction of the suffix.

Strictly speaking, the jurisdiction is the authority that determines the geocode technology and its syntax, and each country has sovereignty over this type of decision — this is already the case e.g. with postal code decisions. It is also suggested to use the subdivision, for example ISO 3166-2:BR define the Brazilian subdivision codes.

As each country is sovereign, it can also define, for example, abbreviations for city names. This hierarchical sequence of countries mnemonic codes is also a spatial hierarchy.



The samples in the table below are all "default geocodes", which should be determined by **jurisdiction**. Each ISO country code prefix indicates the authority that determines the technical standard that defines its local geocode. Only "CV" and "IE" are real jurisdictions, with official fine-grained postal codes after country prefix, that are also standard and valid geocodes.

Jurisdiction	Description	Examples (mnemonic prefix)
BR-SP	This is a Brazilian's ISO 3166-2:BR subdivision, SP is the São Paulo state. The complete prefixes "BR-SP-SP" and "BR-SP-PIR" are not ISO standards. They should be defined by sub-jurisdiction's local conventions, and the local geocodes (the suffixes "5fqf0") results in different geographical locations.	`geo:br-SP-SP-5f.qf0` `geo:br-SP-PIR-5f.qf0`
CV	Cape Verde adopted OLC as official postcode. The ISO 3166-2:CV defines directly its provinces: "CV-PR" is the Praia province and "CV-TA" Tarrafal. The suffixes (e.g. "WF8R+V7") are local geocodes expressed in OLC technology.	`geo:cv-pr-WF8R+V7` `geo:cv-ta-76HX+H6`
FR	France subdivision equivalent to a city is the *commune*, but there are ~36500 communes, many with less than thousand inhabitants; the best ISO 3166-2:FR code is *department*, that have also a name (e.g. "fr-orne" = "fr-61"), equivalents.	`geo:fr-orne-5fqf` `geo:fr-paris-5fqf`
IE	Ireland government defined Eircode as full country geocode. The first part of the Eircode is the routing key (postal district), in the example "D01" for Dublin, "R93" for Carlow and Tullow. The second code is the coordinate.	`geo:ie-R93.E920` `geo:ie-D01.E920`
KH	The ISO 3166-2:KH, of Cambodia defines numeric codes for subdivisions, "17" is Siem Reab and "21" is Taakaev.	`geo:kh-17-3u799m` `geo:kh-21-00222`
RU	As larger country, Russia can use a non-ISO subdivision. Let's suppose the 28 UTM zones (1N, 2N, 34N-60N) reindexed to base32. Prefix and suffixes can use distinct technologies.	`geo:ru-6-3Y7` `geo:ru-V-B0123`

Except by "CV" (Cape Verde) and "IE" (Ireland) of the jurisdiction column in the above table, with suffixes obtained from official geocodes, all other suffixes — e.g. the "5f.qf0" of "geo:br-SP-SP-5f.qf0" or the "00222" of "kh-21-00222" — are only illustrations, there are no associated official geocode system.

Suffixes as location IDs

It is expected as a "natural decision" of a jurisdiction (it is not an obligation), that, in the syntax of the official geocode, after mnemonic prefix, there are a short encoded number that determines the point location in the context of the prefix. This encoded number is indirectally encoding the latitude and longitude.

Appendix-A is explaining the geometrical effect of the use of space-filling curves is the discretization of the latitude-longitude measures into a hierarchical discrete grid. Summarizing: it is impossible to use good "geographical mnemonics" as suffixes, but is possible to use short codes pointing to arbitrary geographical locations as cells of a local grid. Therefore, in this article and its proposal, we are supposing that suffix codes are short codes, and are always interpreted as cell identifiers of a grid.

Objectives

The aim of this article is to describe and to justify the proposal.

Main objectives of the proposal:

1. to define the syntax and syntactic rules to express geocodes;
2. to define the semantic and procedures to interpret geocodes;
3. to standardize the relation between cell area represented by geocode and uncertainty of its centroid;
4. to define labels and abbreviation for geocode technology and for official goecode jurisdictions;
5. to suggest responsibilities and administration of the geocode-type and official namespaces.

Other aim of this article is to dedicate to the public domain the proposal and its “derivatives”, including published algorithms and methodologies, by the same authors (one or more), at *git* platforms.

License

PROPOSAL

This article is presenting a proposal to extend the Geo URI (RFC 5870) syntax and semantics, to support location by geocode.

Syntax

We suggest to change the `geo-path` syntax (sec. 3.3 of the RFC 5870) from

```
geo-path        = coordinates p
coordinates    = coord-a "," coord-b [ "," coord-c ]
```

to

```
geo-path       = coordinates p | geocode p
coordinates    = coord-a "," coord-b [ "," coord-c ]
geocode        = typed-geocode / official-geocode
typed-geocode     = geocode-type ":" suffix
official-geocode = ("via:" official-via / canonic-geocode)
canonic-geocode   = official-prefix "-" suffix
geocode-type   = labeltext *(labeltext / "-")
official-via   = official-prefix "~" via [("-"/"~") suffix]
via            = via-name / via-code
official-prefix   = 2*ALPHA *("-" 1*c-alphanum)
suffix         = c-alphanum
via-code       = c-alphanum
via-name       = 1*(labeltext / "." ) "_" 1*(labeltext / "." )
c-alphanum     = 1*(labeltext / "_" / "." / "~" / "+")
```

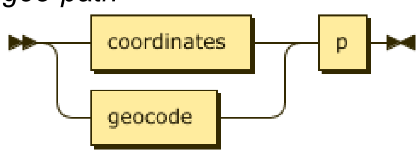
"geo:x:1" is a valid syntax for typed geocode, "geo:x-1" is a valid syntax for official geocode and “geo:1,2” is valid encoding for coordinates. The syntactic inclusions do not affect ordinary ‘geo’ URI or its recommendations.

The official prefix is case insensitive. The suffix, in both uses, is context-dependent, the jurisdiction and/or the geocode type defines more constraints and syntax rules. For human-readability of suffix, it is also suggested the use of base36 or minor (since base4). However base32 is preferred, but it is not an obligation.

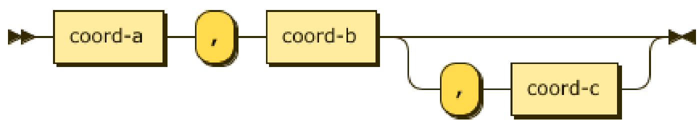
In this proposal, the 'crs' parameter, and all other optional components of the ‘geo’ URI are preserved.

The reserved type “via”, for "geo:via:x" a variation of official geocode, that uses official-prefix to contextualize the name of a *via* (a street, way, road, highway, or any other) and the optional syntax "geo:via:x~y" that transforms the code into a point determined by house number, for addresses.

geo-path =

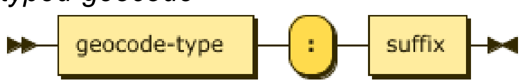


coordinates =

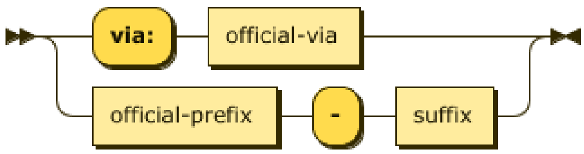


geocode =

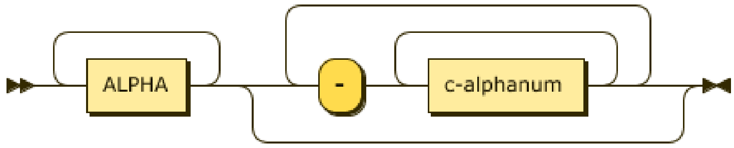
typed-geocode =



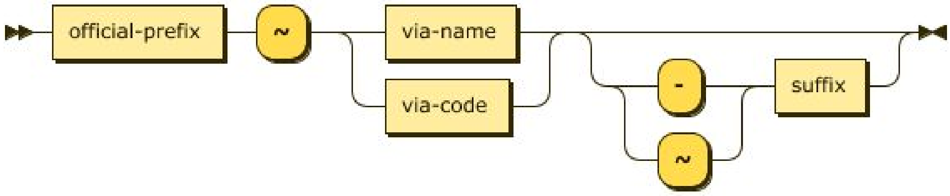
official-geocode =



official-prefix =



official-via =



Secondary syntax changes

One of the general aims of this proposal is to enhance human-readability of the Geo URI.

There are some useful optional parameters that can be included to incentives the use of the uncertainty parameter. This secondary suggestion is to introduce some additional syntax for explicit units or explicit reference to the shape of the nominal uncertainty. It is a change from:

```
uncp          = ";u=" uval
```

to

```
uncp          = ";u=" uval unit
unit          = ["m" / "ucrs" / "m2"]
```

where the unit is optional, and can be changed from (default `m`) meters to the units of the CRS (`ucrs`) or units of the uncertainty region, that is square meters (`m2`).

Semantics

Data contained in a 'geo' URI identifies a physical resource: a spatial location is identified by a geocode or the geographic coordinates, and the CRS is encoded in the URI. A geocode can be an address (*via*), a non-official *global geocode*, or an *official geocode* expressed by a official *mnemonic prefix* and its *local geocode*.

The 'via' geocode-type

In the syntax definition the reserved type “via” designates a second protocol, a dual representation for a point that is also an “official address”, because in many countries must be expressed by both, through:

1. **Official-via** code (complete address): an human-interpretable descriptor of de point, by typical street-name (the *via*) and house number. In the syntax model it is “official-via” and its “suffix”.

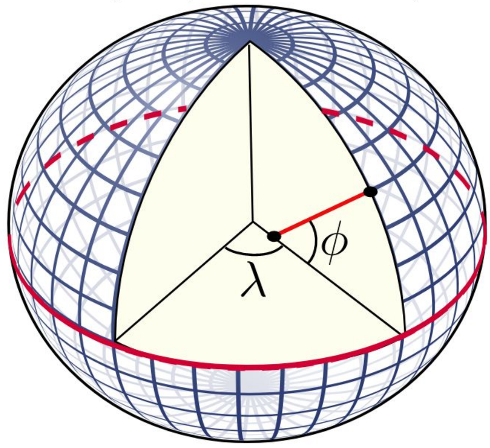
2. Canonic **geocode**: the official *mnemonic prefix* and its *local geocode*.
   The old “classic geocode”, that is a complement of the via code, is the postal code. Here the proposal is to replace old postal codes by canonic geocode, that has the semantic of door location in addresses.

The duplication is intentional, to supply digital uniform expressions for both official locators. The process to translate official-via address to associated point (coordinates or geocode) is known as geocoding. It can not be confused with geo-encoding, the process of encode latitude and longitude into a cell_id or its associated geocode.

The component geocode

A Discrete Global Grid (DGG) is a mosaic which covers the entire Earth's surface. Mathematically it is a space partitioning: it consists of a set of non-empty regions that form a partition of the Earth's surface. [7] The grid of a DGG is a set of cells, and each cell (a simple convex polygon) has its identifier, ***cell-ID***. The [OGC] standard (abstract specification), defines a special class of hierarchical DGG's, the "Discrete Global Grid Systems" (DGGS) that must to offer a rigorous construction from a surrounding polyhedron of the globe, and cells with equal areas at each hierarchical level.

Latitude (**ϕ**) and Longutde (**λ**) on an ellipsoid (default is **WGS-84**)



in a Discrete Global Grid any (λ,φ) coordinate collapses into the center of a grid cell

The geocode is the human-readable code of a *cell-ID* in a DGG. The "geocode-type" (see syntax) specifies the DGG technology and cell-ID rules for representation (e.g. the base32 and the alphabet of this base), by a table fixed by IANA or similar authority. Examples of geocode-types and its possible abbreviations: Geoshash (ghs), UTM zones (utm-zn), OLC (olc).

Sometimes the geocode has a "contextualizer prefix", which is a macro-cell identifier. In the case of official geocodes the contextualizer is the polygon of the country, identified by an ISO country code, and defining the jurisdiction — which will select specific sub-jurisdictions or syntax and semantic rules for the suffix.

To avoid ambiguities, the geographic concept of *jurisdiction* includes *territorial waters*. Another typical source of confusion is "constituent country" (eg. the Netherlands) and "sovereign state, including its territories" (eg. the Kingdom of the Netherlands), when the ISO country code is the same: the preferred one it is the geographically wider concept (eg. the Kingdom).

The geocode's size as uncertainty

Our proposed extension for Geo URI can to encapsulate a geocode, but in legacy systems, where coordinates are expected (or systems where they are preferred), the geocode can be converted.

Theoretically, a Geo URI like "`geo:x,y;u=uval`" can be converted into a URI like "`geo:code`" where the area *a* of the cell, that's the code is representing, is a function of *uval, a=f*(*uval*). This process and assumptions are not precise, but preserves the order of magnitude, for both: geocote to coordinates or coordinates to geocode.

The RFC 5870 text does not formalize the use of the term "uncertainty", and its normative references are also vague with the theme. The text needs a full revision before to extended for conversion rules. We suggest that working groups, at OGC, OSGeo, and OSMF, involved with the subject, express a formal proposal of uncertainty model and valid mathematical simplifications.

For practical use, and for avoid the lost of all uncertainty information, we can adopt, for now, the following considerations:

1. The uncertainty value, `uval`, is a **sum of all uncertainties**, positional (e.g. GPS error) and definitional (e.g. imprecise shape of the object or the probability of being found), and its statistical correlations.
2. It is a "nominal value", a magnitude order, the same valid (after units conversion) for all coordinates and location probability. See section 3.4.3 of the RFC, "The single uncertainty value is applied to all dimensions given in the URI".
3. Can be **modeled by spherical volume** around the point (**or a disk** of horizontal projection). This volume

[7] See Wikipedia and Sahr et al. (2003)

is can be interpreted as the set of positions where the object can be found at a confidence of 95%. See the normative reference, RFC 5491, sec. 5, “It is RECOMMENDED that uncertainty is expressed at a confidence of 95% or higher”.

4. The uncertainty sphere can be **projected onto the grid of the geocode**, resulting in a unique cell nominally containing 68% (2σ) or more of the projected region. It is a nominal value; and it is less tham 95% (3σ) because the process of “snap to grid and select one cell” reduces confidence region.

Geometrically, we are assuming that the centroid of the grid cell (represented by the geocode) is the mean location and uncertainty is represented by the cell area, *a*. Therefore, using the considerations, we can suppose:

$$a = \pi \cdot \sigma^2 \;\Rightarrow\; \sigma = \sqrt{(a/\pi)}; \qquad uval = 2\sigma \;\Rightarrow\; \mathbf{uval = 2\sqrt{(a/\pi)}} \; = 2\cdot(\sqrt{a})\cdot\sqrt{(1/\pi)} \;\sim 1.1284\cdot\sqrt{a}$$

The last bolded equation shows a reasonable nominal relationship between uncertainty *uval* and the area *a* of the cell. Its mathematically valid for both, *uval*(*a*) and its inverse function *a*(*uval*).

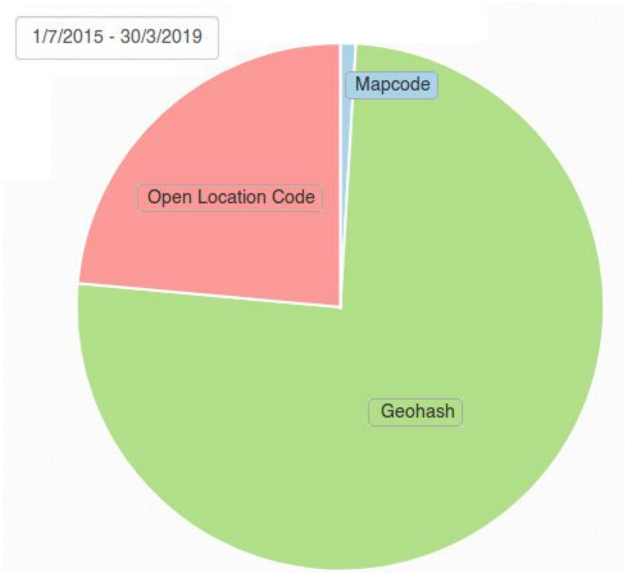
For example the Geohash cell "`6gyf4bf1`" is a rectangle with 670 m² of area, which can be approximated to a 26 m square side, having uncertainty *uval*(670 m²)=29 m. The prior level, a larger cell "`6gyf4bf`", is 21424 m² and *u*(21424 m²)=165 m. For this scale and precision, they are distingible uncertainties, therefore inversible.

Rule, when geocode use “u” parameter: if calculated *u*(*a*) greater than supplied *uval*, use *u*(*a*), else use *uval*.

Geocode technologies selection

The ‘geo’ URI is a *open* protocol. The DGG technologies, like Geohash, S2 geometry or OLC, to be used in a ‘geo’ URI must be also implemented by open-source algorithms.

There are many DGGs in use, and many that are not so reliable. Some quality and “comparative index of use” can be used to increment quality and reduce diversity. For example the illustration side compares the number of pageviews of the Geohash page in Wikipedia and the supposed two other popular geocodes, OLC and Mapcode.



	Page title	Views
	Geohash	31.541
	Open Location Code	20.335
	Mapcode	2.114
	3 pages	53.990

Another selection criteria may be the “at least one country is using”, or one international SDI body like INSPIRE, GeoSUR, etc.

The DGG standardization is a “natural standard” for a country (official DGG). Goodchild (2001) demonstrated that standardized DGGs ensures integrated source of knowledge for researchers, developers, agencies, and corporations. A standard DGG has central role georeferencing, indexing, and discretization contexts.

We suggest that IANA with organizations like OSFM and OSGeo evaluate registration requests.

Jurisdiction and governance of the official geocodes

This proposal is adopting the also named "self‑sovereignty" of official geocodes and its jurisdiction hierarchy, that is defined by [DID-2019draft] as an architecture of IDs "*should give entities, both human and non-human, the power to directly own and control their digital identifiers without the need to rely on external authorities*".

The governance of the namespaces and syntactic rules of official geocodes are hierarchically determined. The authority of the prefix root level, the country of the ISO-country-code, determines the country's DGG-reference and the existence or not of an hierarchy of authorities (by sub-prefixes).

Context resolution

In an array of URIs, the **string concatenation** can be used avoid redundancy. It is not part of the protocol, is only a naming recommendation: the prefix of the array use the label "**geo_pref**". Example for JSON object:

```
{"geo_pref:br-SP-PIR-4":["23","25","ab"]}
```

will be interpreted as `["geo:br-SP-PIR-423", "geo:br-SP-PIR-425", "geo:br-SP-PIR-4ab"]`.

When the relation with the *geo_pref* is not implicit, use the label "**geo_suff**" at suffixes. Example:

```
{"geo_pref":"br-SP-PIR-4","x1":true,"x2":[{"id":1,"geo_suff":"2340"},{"id":2,"geo_suff":"2501"}]}
```

In a map visualization the label name can be hidden by Set Theory metaphor for *contains*, and compatible colors for visual string concatenation.

Context resolver for interfaces

The geocode is to be used as it, without transformations. Before capturing, building or editing, it is possible to use a "context-reference geo URI", *r*, that can supply the official-prefix or geocode-type (see Syntax). This is a suggestion for interface syntax and expansion rules.

Suppose r = "geo:br-SP-PIR-456", a geocode that uses Geohash (ghs) technology, syntax rules:

- geocode-type: empty or ".". Example, "geo::123", will be expanded to "geo:ghs:123".
- official-geocode starting with ".". Example, "geo:.123", will be expanded "geo:br-SP-PIR-123".

A typical contextualization scenario is a mobile device with GPS and its user editing a chat.

REFERENCES

K. Sahr, D. White, A .J. Kimerling (2003). "*Geodesic discrete global grid systems*". urn:doi:10.1559/152304003100011090.

M. F. Goodchild (2001), "*Discrete Global Grids for Digital Earth*". urn:doi:10.1.1.83.9660.

N. Jaklin, R. Geraerts (2016), "Navigating Through Virtual Worlds: From Single Characters to Large Crowds". urn:doi:10.4018/978-1-4666-9629-7.

Standards

[LexML-BR] - Brasil (2008), "*LexML Brasil, Parte 2 – LexML URN*", http://projeto.lexml.gov.br/documentacao/Parte-2-LexML-URN.pdf

[OGC] - Open Geospatial Consortium (2017), "*Topic 21: Discrete Global Grid Systems Abstract Specification*". Document 15-104r5 version 1.0 (HTML).

[RFC 4180] - Y. Shafranovich (2005), "*Common Format and MIME Type for Comma-Separated Values (CSV) Files*". urn:rfc:4180

[RFC 5031] - H. Schulzrinne (2008), "*A Uniform Resource Name (URN) for Emergency and Other Well-Known Services*". urn:rfc:5031

[RFC 5491] - J. Winterbottom et. al. (2009), "*GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations*". urn:rfc:5491

[RFC 5870] - A. Mayrhofer, C. Spanring (2010), "*A Uniform Resource Identifier for Geographic Locations ('geo' URI)*". urn:rfc:5870

[RFC 7159] - T. Bray (2014), "*The JavaScript Object Notation (JSON) Data Interchange Format*". urn:rfc:7159

[WGS84] - National Imagery and Mapping Agency, "*Technical Report TR 8350.2, World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems*". Last edition EGM2008.

Drafts of standards

[DID-dft19] - D. Reed et. al. (2019) - "*Decentralized Identifiers (DIDs), Data Model and Syntaxes for Decentralized Identifiers (DIDs)*", https://w3c-ccg.github.io/did-spec

[URN-LEX-dft18] - P. Spinosa, E. Francesconi, C. Lupo (2009-2018), "*A Uniform Resource Name (URN) Namespace for Sources of Law (LEX)*". https://datatracker.ietf.org/doc/draft-spinosa-urn-lex

Sources of illustrations and examples

- IBGE (2019) "Censo Agropecuário de 2017", ftp://ftp.ibge.gov.br/Censo_Agropecuario/Censo_Agropecuario_2017/Cadastro_Nacional_de_Enderecos_Fins_Estatisticos
- ODI (2016a) - "OpenDataIndex/Locations", https://index.okfn.org/dataset/postcodes
- ODI (2016b) - "OpenDataIndex/Land Ownership", https://index.okfn.org/dataset/land

APPENDIX

A. Hierarchical geocode semantic

According to the [OGC] standard the main benefit of a “DGG System” (DGGS) is to provide spatial reference system (for storing and integrating information), focussing on area-based applications. To be a DGGS, all the cells of the grid (of some level) must represent equal areas, resulting in an area-preserving reference system.

For typical use of Geo URI, illustrated by [RFC 5031], persons with its mobile phones (they are like GPS users) in a specific city, the city’s grid at 5 meter scale is area-preserving. So, with some adjustments, non-OGC grids can satisfy the OGC requirements, behaving as a DGGS. We can suppose that some jargon used in this article have equivalent terms at OGC standard:

OGC standard	**non-OGC (article’s jargon)**		**OGC standard**	**non-OGC (article’s jargon)**
DGGS cell	DGG cell		geo-encoding	assigning geocode or cell_ID
cell refinement	partition or “next level”		encoded cell address	cell_ID
DGGS resolution	hierarchical level		tessellation or grid	grid

The OGC standard enable cell address (cell_id) to start with a polyhedra face identifier. We can generalize to cell_id0, that can be a face or a jurisdiction prefix, and suppose cell_id = <cell_id0,cell_id_suffix>.
PS: in general we can ignore cell_id0, and suppose cell_id and cell_id_suffix as synonymous.

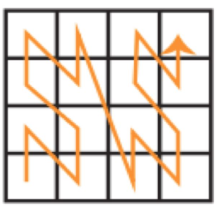
The **geocode** concept, as human-readable (and short) representation of the cell_id_suffix, was not defined in the [OGC] standard, that use many terms relating to geo-encoding, as “geodetic identifier", “encoded cell address”, “index” and “geographic identifier”.

Examples of geo-encoding (encoding latitude and longitude)

The most simple strategy to encode two coordinates into only one number is concatenating, for example *a*=`72` and *b*=`19` into `7219`. To compress we can encode into base36 or base32 resulting in “71J” reducing in ~30% the number of digits of usual coordinates. If we make this process a standard, we can say that “71J” was geo-encoded. But this simplest code have two main problems:
1) to decode need a fixed number of digits in *a*, e.g. if *a*=`7` and *b*=`219` the concatenation is the same;
2) it is not possible to compare, checking similarity. Ideally, when two points in its grid are neighbors, is expected that the encoded form result in similar numbers (low unsigned difference).
Suppose e.g. a second pair “`73` and `18`” encoded as `7318`, not seems similar to `7219`.

A solution, perhaps the the most “simple and good” strategy, generating hierarchical codes, is to interlacing digits. For example “`72` and `19`” can be interlaced as `7129`. Now the similar pair (`73` and `18`) is interlaced as `7138`, and seems similar — they have same prefix, 71.
To do better (bigger common prefix), optimizing hierarchy and the computational mechanism, we interlace digits of the binary representation of the two input numbers:

```
     a = 0  1  0  0  1  0  0  0  = 01001000 = decimal 72
     b = 0  0  0  1  0  0  1  1  = 00010011 = decimal 19
result = 00 10 00 01 10 00 01 01 = 0010000110000101 = decimal 8581
```



11	13	31	33
10	12	30	32
01	03	21	23
00	02	20	22

Comparing with the pair “`73` and `18`” (in binary “1001001” and “0010010”) results `8582`, a neighbor in the grid.

The geometrical effect of interlacing latitude and longitude is a fractal, knowd as Z-order curve (side illustration). When we replace the interlacing procedure by more complex algorithms, we can obtain many other fractal space-filling curves (e.g. the famous Hilbert curve). To express the binary result into a compact and human-readable representation, the most usual is to convert the number into base16 or base32 notations. Generalizing, the geometrical effect of the use of space-filling curves is the discretization of the latitude-longitude measures into a hierarchical discrete grid.

B. Compatibility with other RFCs

Supplementary material. Examples of use of this proposal in other RFCs, demonstrating that the proposal can be easily merged into the RFC ecosystem, and solves old problems. Important motivations are postal code modernization and formal land demarcation, with open formats and open protocols (see [ODI16a], [ODI16b]).

Extending RFC 5031 with official geocodes

> "*The content of many communication services depends on the context, such as the user's location. We describe a 'service' URN that allows well-known context-dependent services that can be resolved in a distributed manner to be identified. Examples include emergency services, directory assistance, and call-before-you-dig hot lines*", [RFC 5031].

The URN's `urn:service:sos` and `urn:service:sos.animal-control` are valid URNs, but has some problems:

- use english, is not multilingual: language-neutrality is an especially important feature that will promote adoption of the standard by organizations that must adhere to official-language requirements of each county.
- requires centralized attendant: the standard ignores the cost of centralization, and most official third-world communication services are not centralized. A generic name "SOS" can't be used without relating to a local authority (in general municipality).
- is context-dependent: who needs to call SOS also needs to say "I am here". The standard assumes that the URN-solver mechanism also has a secure geographic localization mechanism, which, precisely in emergencies, is not always true.

A simple solution is to extend the "urn:service" for third-world and non-English world. The extension not affect legacy resolvers, and open doors for new users. Changes "service-URN" syntax to:

```
service-URN  = "URN:service ":" [officialGeocode ";"] service
```

And the `service` names are also changed to translations compatible with the official country's language.
The officialGeocode can use any degree of detail, which is guaranteed by the hierarchical nature of the official geocodes, as described in this article, organized in hierarchical mnemonic prefix and hierarchical suffix point-locator.

Commented examples:

Example	Comment
`urn:service:sos.animal-control`	An example of sub-service for the 'sos' service, at sec. 4.2 of RFC 5031.
`urn:service:us;sos.animal-control`	The "SOS USA" service determined without context.
`urn:service:br;sos.controle-animal`	The same animal-control service, but at Brazilian jurisdiction, using correct language.
`urn:service:br-sp-pir;sos.controle-animal`	Avoiding need for spatial resolution, to the correct authority, because in Brazil there are not centralized.
`urn:service:br-sp-pir-1234;sos.ambulancia`	It not exists, but imagining that "br-sp-pir-1234" is a valid official Brazilian geocode. This URN can be used for both, request data from the nearest SOS service, or indicate that there is demand for the service at this location.

Extending RFC 7159 with hierarchical geocodes

The GeoJSON format is perhaps the most successful "geographical RFC", the main advantages other open formats (like SHP, WKT, KML and others) are its lightness, universality and simplicity. But can be more light if reuses information about context or hierarchy of geographical coordinates, and a compact geocode.

Supposing the use of geocodes (e.g. Geohashes) instead coordinates in an adapted GeoJSON, and the use of hierarchical relations, through the "geo_pref" field, to reduce geocode lengths. It reduces memory consumption and, most important, represent a gain in readability. It is more easy, for humans, to read short codes instead long ones, and more easy to compare two geocodes tham four coordinates. It is possible to use it (the geocodes and GeoJSON) to express more readable official transcriptions (example).

To compare by real test case with [IBGE 2019] data, we can do direct use of geocodes, without no adaptation, by the CSV open format, as in [RFC 4180]. The samples has 188591 rows.

coordinates.csv	**geoshashes.csv**	**cut_*.cvs**
Usual file.	Global geocodes.	Local geocodes. Splitting *geoshashes.csv* in ~100 files named by prefix: *cut_6u5s.csv*, *cut_6u5t.csv*, etc.

`latitude,longitude,status,Z`
`-21.6586217,-51.0844783,1,460`
`-21.6985600,-51.0749883,1,426`
`-21.5832850,-51.0731000,2,384`
`...`
	`geohash,status,Z`
	`6u5sv91mmkk,1,460`
	`6u5stcbkdx0,1,426`
	`6u5tjycsq80,2,384`
	`...`
		`geohash,status,Z`
		`v91mmkk,1,460`
		`tcbkdx0,1,426`
		`...`
		`geohash,status,Z`
		`jycsq80,2,384`
		`...`

file size: 5.8 Mb	file size: 3.3 Mb (61% of coordinates)	file size sum: 2.6 Mb (48% of coordinates). Note: ~70 files with prefixes of 4 characters and ~30 files with 3 characters.
Zip file: 1.87 Mb	Zip file: 1.45 Mb	Zip file: 1.38 Mb

In the zipped files the gain is ~22% of file size reduction (~26% when cut prefixes of hierarchical groups), no significative. The rationale for geocodes is mainly for human readability and hierarchical grouping.

For non-zipped format (human-readable) the reduction is ~50%. It is a strong indicator for geocode adoption in land demarcation reports, land demarcation contracts and law (example of official transcription), enhancing transparency by text reduction, uncertainty description and clustering blocks of land demarcation.

Harmonizing URN-LEX with geocodes

The proposed mnemonic prefixes of official geocodes, that have also semantic of jurisdiction, can be used as it or adapted to the Brazilian standard URN LEX of [LexML-BR], that is also the RFC proposal [URN-LEX-dft18].

For example "`urn:lex:br;sao.paulo;campinas:municipal:decreto:2008-01-24;16135`" can be reduced to "`urn:lex:br-sp-cam:municipal:decreto:2008-01-24;16135`" and interoperate by jurisdiction prefix "`urn:lex:br-sp-cam`", using semantic equivalence with "`geo:br-sp-cam`".